Image: Mirror Neuron System (MNS) in humans.

More Neural Language Models

CSC401/2511 – Natural Language Computing – Winter 2023

Raeid Saqur

Lecture 7

University of Toronto

Logistics

- Office hours: Wed 12.30 – 1.30 pm (over zoom, note the channel)
- A2: due Mar 10, 2023 – *errata recap*.
- A2 tutorials planned schedule:
  - ~~Feb 17: A2 tutorial – 1~~
  - Mar 3: A2 tutorial – 2 (ft. Frank Niu)
  - Mar 10: A2 – Q/A and OH (*submission due at mid-night*)
- A3: release Mar 11, 2023
- Final exam: date to be finalized soon
- Lecture feedback:
  - Anonymous
  - Please share any thoughts/suggestions
- **Questions?**

Scan Me

More Neural Language Models

Lecture plan for today (L7 – 1/1)

- Emergent NLM architectures:
  - Encoder only (BERT, BERTology findings)
  - Encoder-Decoder: unified text-to-text format (T5)
  - Decoder only auto-regressive models (GPT):
    - covered in detail at a later lecture (L13)
  - Token-free models:
    - Importance, and the whys
    - Selective example: CANINE
- Trends in Neural Language Models
  - Scaling laws of NLMs
  - NLMs as foundation models & implications

BERT: Bidirectional Encoder Representations from Transformers

Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP
1	T5 Team - Google	T5		89.7	70.8	97.1	91.9/89.2	92.5/92.1	74.6/90.4
2	ALBERT-Team Google Language	ALBERT (Ensemble)		89.4	69.1	97.1	93.4/91.2	92.5/92.0	74.2/90.5
+ 3	王玮	ALICE v2 large ensemble (Alibaba DAMO NLP)		89.0	69.2	97.1	93.6/91.5	92.7/92.3	74.4/90.7
4	Microsoft D365 AI & UMD	FreeLB-RoBERTa (ensemble)		88.8	68.0	96.8	93.1/90.8	92.4/92.2	74.8/90.3
5	Facebook AI	RoBERTa		88.5	67.8	96.7	92.3/89.8	92.2/91.9	74.3/90.2
6	XLNet Team	XLNet-Large (ensemble)		88.4	67.8	96.8	93.0/90.7	91.6/91.1	74.2/90.3
+ 7	Microsoft D365 AI & MSR AI	MT-DNN-ensemble		87.6	68.4	96.5	92.7/90.3	91.1/90.7	73.7/89.9
8	GLUE Human Baselines	GLUE Human Baselines		87.1	66.4	97.8	86.3/80.8	92.7/92.6	59.5/80.4
9	Stanford Hazy Research	Snorkel MeTaL		83.2	63.8	96.2	91.5/88.5	90.1/89.7	73.1/89.9
10	XLM Systems	XLM (English only)		83.1	62.9	95.6	90.7/87.1	88.8/88.2	73.2/89.8

BERT

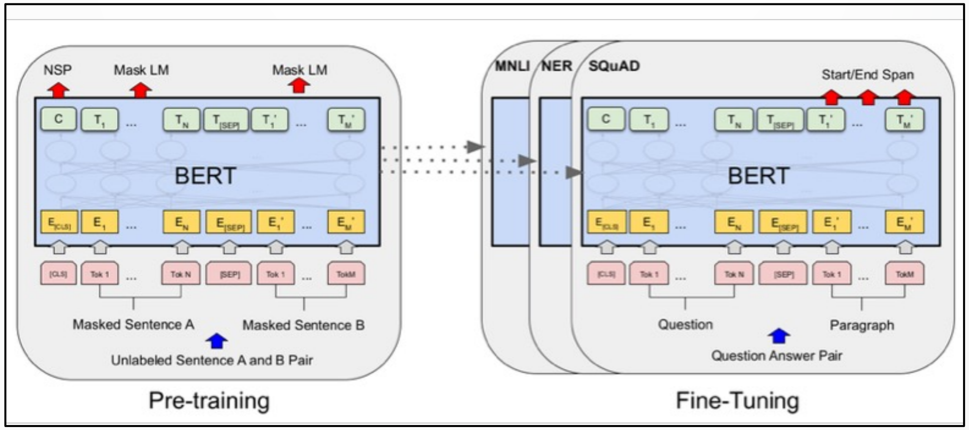
Humans

- The age of humans is over?

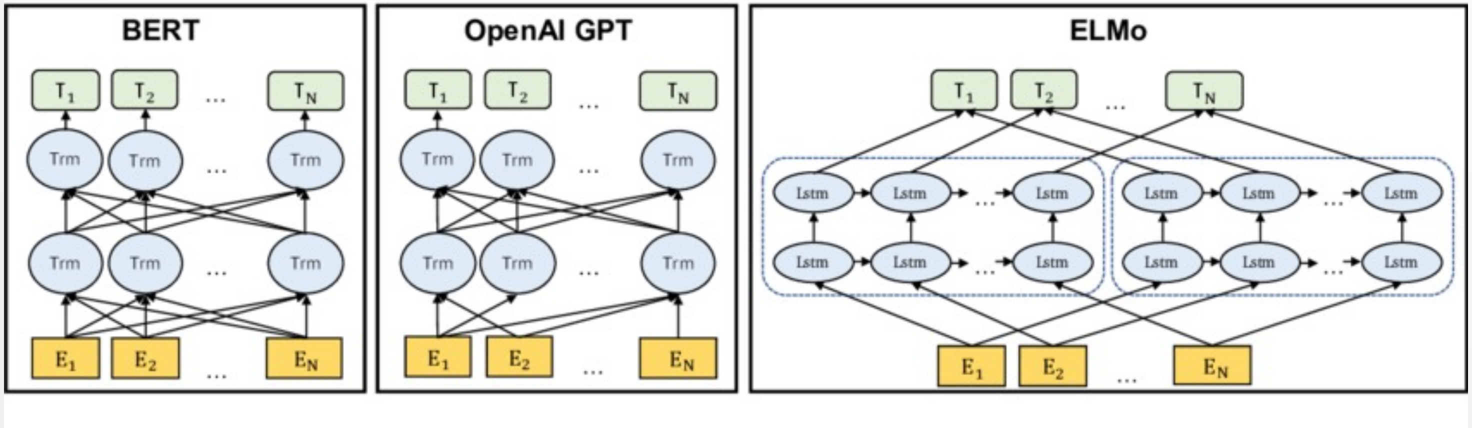
BERT: Bidirectional Encoder Representations from Transformers

Think of the **encoder** part of the transformer architecture

- Landmark, pivotal neural LM that has become an ubiquitous baseline in NLP.
- BERT is conceptually simple (multi-layer, bidirectional transformer), empirically powerful.



- Unlike predecessors (ELMo) or contemporaneous LMs (GPT), BERT is deeply **bi**directional and independent of task-specific features with unified architecture across different tasks.



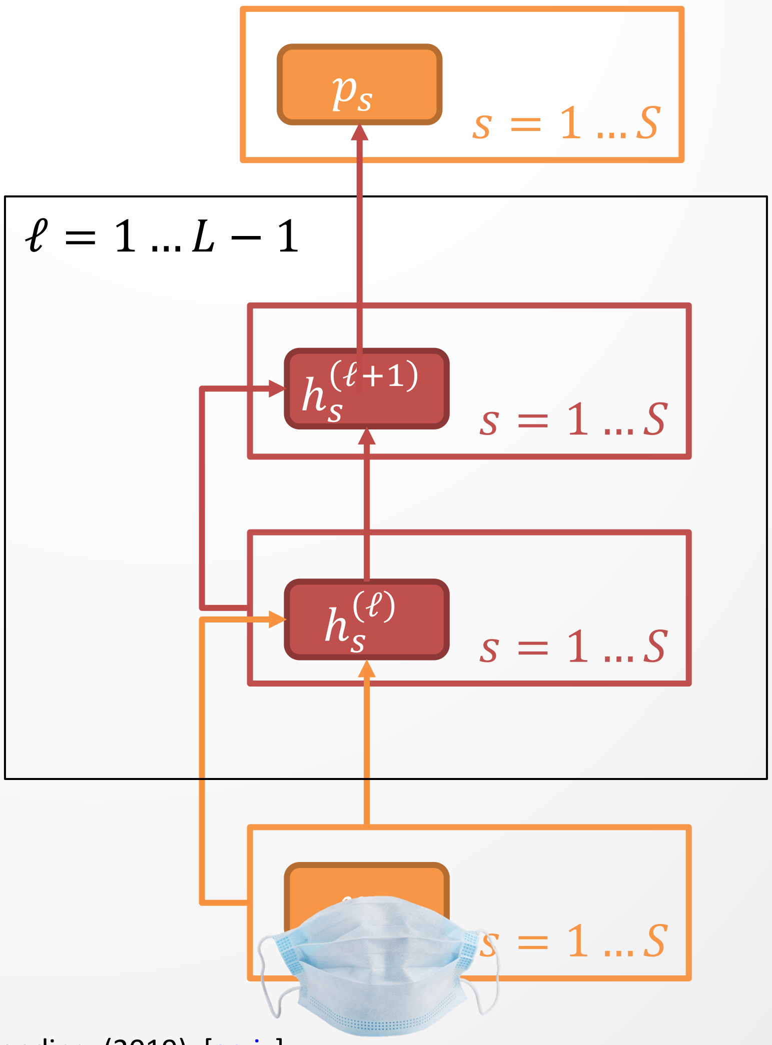
Devlin *et al.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2019). [arxiv]

Code and models: https://github.com/google-research/bert [Colab] Google AI

UNIVERSITY OF TORONTO

BERT: Bidirectional Encoder Representations from Transformers

- First, **pre-trained** on (large) unlabeled data on two unsupervised tasks/objectives:
  - Masked LM (**MLM**), and
  - Next Sentence Prediction (**NSP**)
- Then, **fine-tuned** using labeled data from downstream tasks
- Training entails feeding the final hidden vectors to an output FFN layer with softmax over the possibilities (e.g. the vocabulary as in a standard LM)



Devlin *et al.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2019). [arxiv]

Code and models: https://github.com/google-research/bert [Colab] Google AI

UNIVERSITY OF TORONTO

BERT: Bidirectional Encoder Representations from Transformers

Pre-training objectives

- Masked LM (**MLM**): predict randomly masked words:

Input: The man went to the [MASK]$_1$. He bought a [MASK]$_2$ of milk .
Labels: [MASK]$_1$ = store; [MASK]$_2$ = gallon

  - 80% of the target words are masked with: [MASK]. 10% are replaced with another word, and 10% are kept as-is, to bias '*towards the observation*'.
  - *Variants*: masking granularity can be varied (word-piece, word, span) with respective quirks. E.g., masking named entities improves structured knowledge representation.

- Next sentence prediction (**NSP**): does sentence B follow A?

Sentence A = The man went to the store. **Sentence B** = He bought a gallon of milk. **Label** = IsNextSentence	**Sentence A** = The man went to the store. **Sentence B** = Penguins are flightless. **Label** = NotNextSentence

  - 50% of the time true, 50% of the time it's a random sentence.
  - Later research finds removing the NSP task does not hurt, or slightly improves performance. [2]

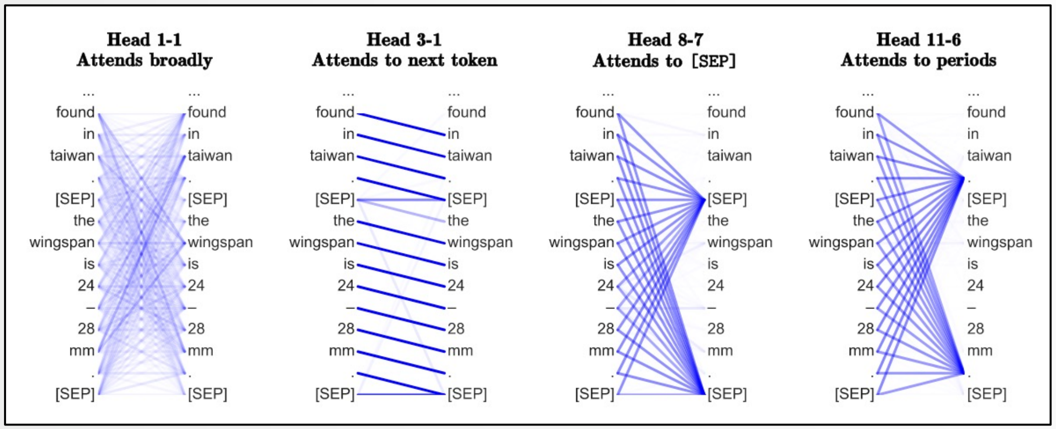
[1] Aroca-Ouellette S, Rudzicz F (2020) On Losses for Modern Language Models, EMNLP.
[2] Rogers, Anna et al. "A primer in BERTology: What we know about how BERT works." TACL(2020). link

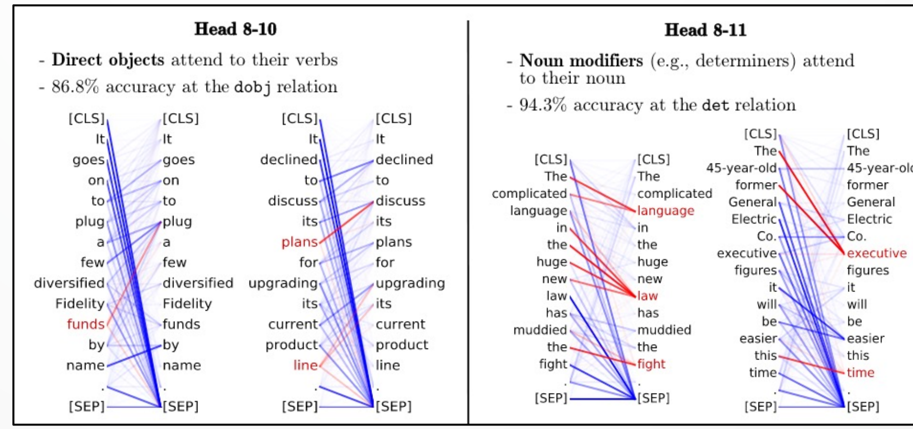
UNIVERSITY OF TORONTO

BERT: Bidirectional Encoder Representations from Transformers

Findings from ablative studies [1,2,3]

- **Heads**: Analysis of the multi-headed attention mechanism in BERT shows attention heads exhibiting attentions on various linguistic (e.g. syntax, coreference) patterns. [1]





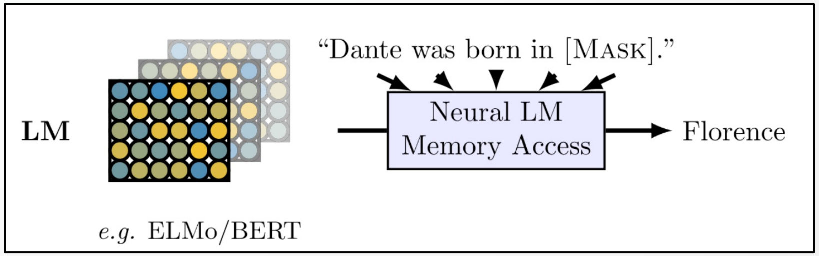
- **Layers**: linear word order and surface features captured most by lower layers. Syntactic information most prominent in middle layers. Semantic and task specific features are best captured in higher/final layers.
- Research on proposed improvements and modifications to BERT, both architectural choices (e.g. # of layers, heads) and training methods is voluminous and ongoing. Due to overall trend towards larger model sizes, systematic ablations have become prohibitively expensive.

1. Clark et al. "What does bert look at? an analysis of bert's attention." (2019). link
2. Tenney et al. "BERT rediscovers the classical NLP pipeline." (2019). link
3. Rogers, Anna et al. "A primer in BERTology: What we know about how bert works." TACL(2020). link

BERT: Bidirectional Encoder Representations from Transformers

Findings from ablative studies

- **Limitations:** BERT's possession of impressive syntactic, semantic, and world knowledge has caveats.
- **World Knowledge:**
  - BERT struggles with pragmatic inference, and role-based event knowledge.
  - It can 'guess' object affordances and properties, but cannot reason about relationships between them. Example: it 'knows' people can walk into houses, houses are big, but cannot infer that houses are bigger than people.
- **Semantic Knowledge:**
  - Struggles with representations of numbers.
  - Surprisingly brittle to *named entity* replacements: e.g. 85% drop in performance in coreference task with names replaced.

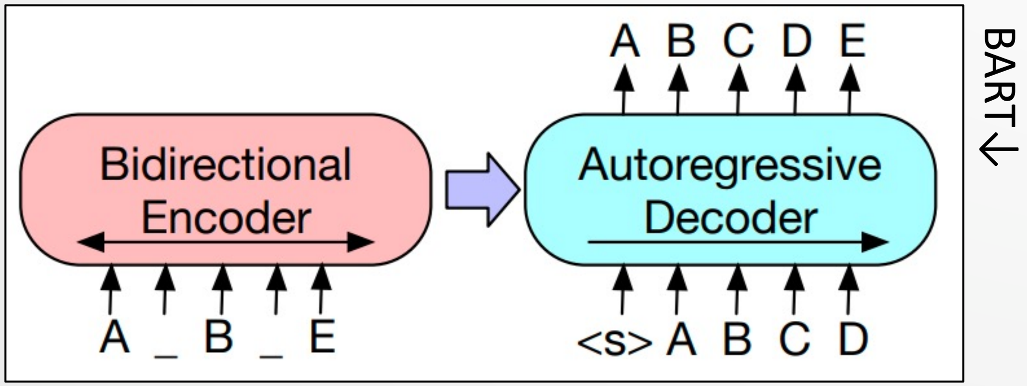


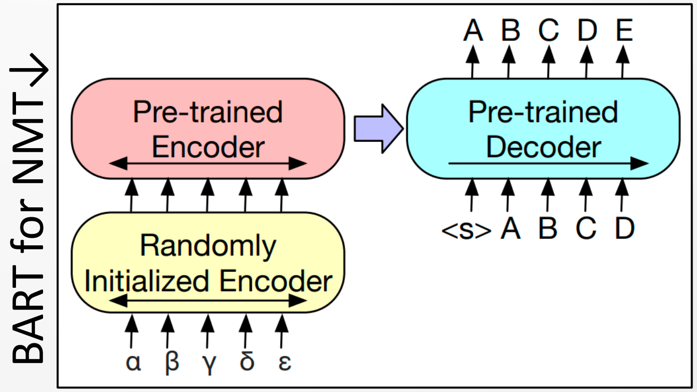
- **Syntactic Knowledge:**
  - Does not 'understand' negations and is insensitive to malformed input.
  - Findings suggest that either its syntactic knowledge is incomplete, or not dependent on it for solving its tasks.

UNIVERSITY OF TORONTO

Aside – BERT → BART → NMT

- Explosion of variants to BERT
- Pretrained BERT language model used to re-score/fine-tune downstream NLP tasks
- BART (Lewis *et al*, 2020) adds the decoder back to BERT, keeping the BERT objective
- Add some source language layers on top to train for NMT





Lewis, Mike, et al. "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension." (2019). link.

UNIVERSITY OF TORONTO

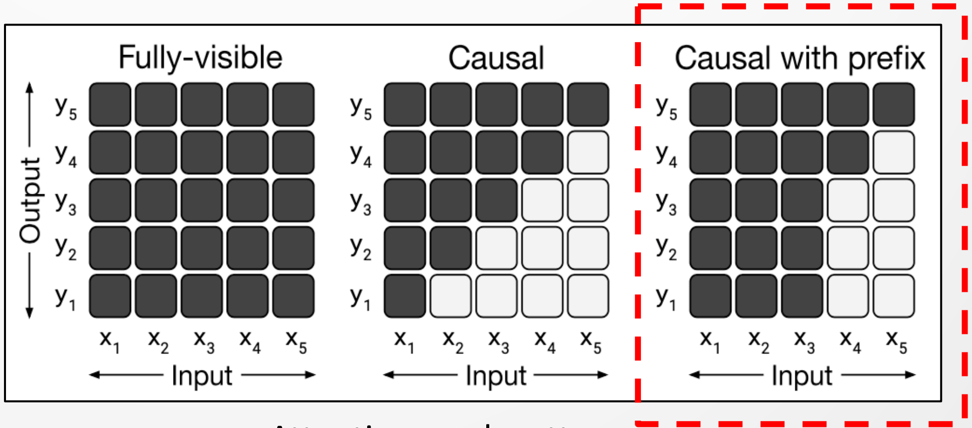
T5: Text-to-Text Transfer Transformer

A *refined* Transformer updated with better methodologies

- T5 is an unified framework that casts all NLP problems into a '*text-to-text*' format.
- Architecturally (almost) identical to the original Transformer (Vaswani et al., 2017).
- Draws from a systematic study comparing pre-training objectives, architectures, unlabeled data sets, transfer approaches, and other factors on dozens of language understanding tasks.
- Introduces and uses a new curated dataset: "*Colossal Clean Crawled Corpus*" (C4) for training.

Distinguishing features:

- Consistent, task-invariant MLE training objective.
- Self-attention "mask" with prefix.
- Unsupervised "denoising" training objectives: *span corruption* (conceptually same to MLM, mask 'spans' instead of words).



Attention mask patterns

1. Raffel et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." (2020). link

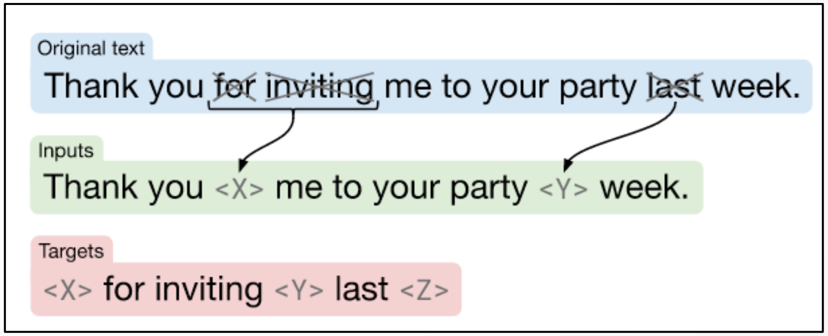
T5: Text-to-Text Transfer Transformer

Example Task: English to German (En-De) translation:

Input sentence: "*That is good.*"
Target: "*Das ist gut.*"

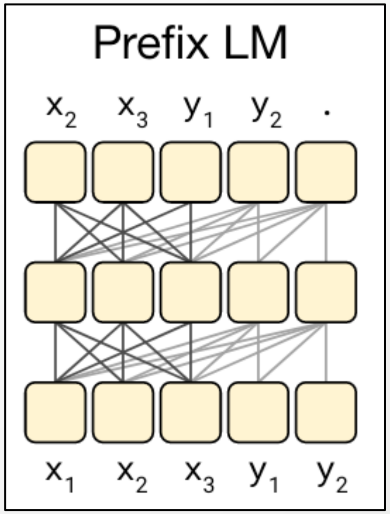
- **Training:** task specification is imbued by prepending task prefix to the input sequence. Model trained on next sequence prediction over the concatenated input sequence:

  "*translate En-De: That is good. Das ist gut.*"



Input/Output format for training denoising objective

- For prediction, the model is fed **prefix**:
  - "*translate En-De: That is good. target:*"
- For **classification** tasks, the model predicts a single word corresponding to the target label.
- E.g. MNLI task of entailment prediction:
  - "*mnli premise: I hate pigeons. hypothesis: I am hostile to pigeons. entailment.* "
- Model predicts label: {"entailment", "neutral", "contradiction"}.



UNIVERSITY OF TORONTO

The Open AI GPT papers

- The GPT papers:
  - GPT (2018)
  - GPT2 (2019)
  - GPT3 (2020)

- Each builds on the predecessor
- Auto-regressive, unidirectional (*left to right*) architecture
- Detailed discussion in lecture 13: LLMs

Improving Language Understanding by Generative Pre-Training

Alec Radford OpenAI alec@openai.com
Karthik Narasimhan OpenAI karthikn@openai.com
Tim Salimans OpenAI tim@openai.com
Ilya Sutskever OpenAI ilyasu@openai.com

Language Models are Unsupervised Multitask Learners

Alec Radford [*1] **Jeffrey Wu** [*1] **Rewon Child** [1] **David Luan** [1] **Dario Amodei** [**1] **Ilya Sutskever** [**1]

Language Models are Few-Shot Learners

Tom B. Brown[*] Benjamin Mann[*] Nick Ryder[*] Melanie Subbiah[*]
Jared Kaplan[†] Prafulla Dhariwal Arvind Neelakantan Pranav Shyam Girish Sastry
Amanda Askell Sandhini Agarwal Ariel Herbert-Voss Gretchen Krueger Tom Henighan
Rewon Child Aditya Ramesh Daniel M. Ziegler Jeffrey Wu Clemens Winter
Christopher Hesse Mark Chen Eric Sigler Mateusz Litwin Scott Gray
Benjamin Chess Jack Clark Christopher Berner
Sam McCandlish Alec Radford Ilya Sutskever Dario Amodei

OpenAI

GPT: model & architecture

- Architecture evolution: GPT3 ← GPT2 + mods ← GPT + mods
- Core architecture follows classic 'language modeling':

$$p(x) = \prod_{i=1}^{n} p(s_n | s_1, ..., s_{n-1})$$

- Learning to perform a task as estimating distribution *P(output | input)*
- Original GPT[1] trains a standard LM objective to maximize the likelihood:

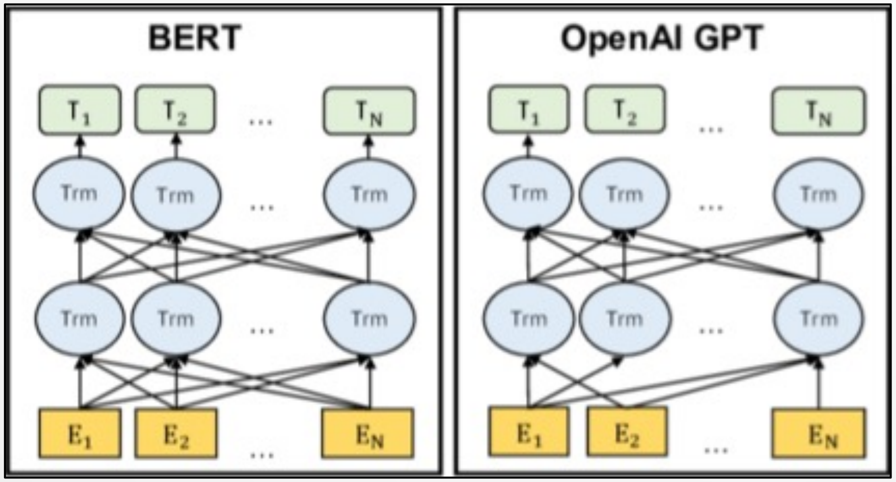
$$L(\mu) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta)$$

  - Given an unsupervised corpus of tokens $\mathbf{\mu} = \{\mu_1, \dots, \mu_n\}$, where k is context window, P is modelled using a neural network with parameters θ
- GPT uses a multi-layer Transformer ***decoder*** for the language model

[1] Radford, Alec, et al. "Improving language understanding by generative pre-training." (2018).

Key architectural differences

- GPT vs. BERT-variants:
  - GPT uses 'transformer' blocks as *decoders*, and BERT as *encoders*.
  - Underlying (block level) ideology is same
  - GPT (later Transformer XL, XLNet) is an **autoregressive** model, BERT is not
    - At the cost of auto-regression, BERT has bi-directional context awareness.
  - GPT, like traditional LMs, outputs (predicts) one token at a time.
- Compare with T5, BART that uses encoder-decoder



[1] Radford, Alec, et al. "Improving language understanding by generative pre-training." (2018).

Token free models

- Unlike the ubiquitous pre-trained LMs that operate on sequences of tokens corresponding to word or sub-word units, *token free models*:
  - ⊕ Operate on raw text (bytes or characters) **directly.**
  - ⊕ Removes necessity for (error-prone, complex) text preprocessing pipelines.
  - ⊖ Con: raw sequences significantly **longer than token sequences**, increases computational complexity. (Reminder: *'attention' costs are quadratic to the length of input sequence*)
- Pitfalls of explicit (word, sub-word) tokenization:
  - Need for large language dependent (fixed) vocabulary mapping matrices.
  - Applies hand-engineered, costly, language-specific string tokenization/segmentation algorithms (e.g. BPE, word-piece, sentence-piece) requiring linguistic expertise.
  - Heuristic string-splitting, however nuanced, cannot capture full breadth of linguistic phenomena, (e.g. morphologically distant agglutinative, non-concatenative languages). Other examples include languages without white-space (Thai, Chinese), or that uses punctuation as letters (Hawaiian, Twi). *Fine-tuning* tokenization needs to match *pretraining* tokenization methods.
  - Brittle to noise, corruption of input (typos, adversarial manipulations). Corrupted tokens lose vocabulary coverage.

1. Clark et al. "**CANINE**: Pre-training an efficient tokenization-free encoder for language representation." (2021). link
2. Xue et al. "**ByT5**: Towards a token-free future with pre-trained byte-to-byte models." (2022). link

Token free models - CANINE

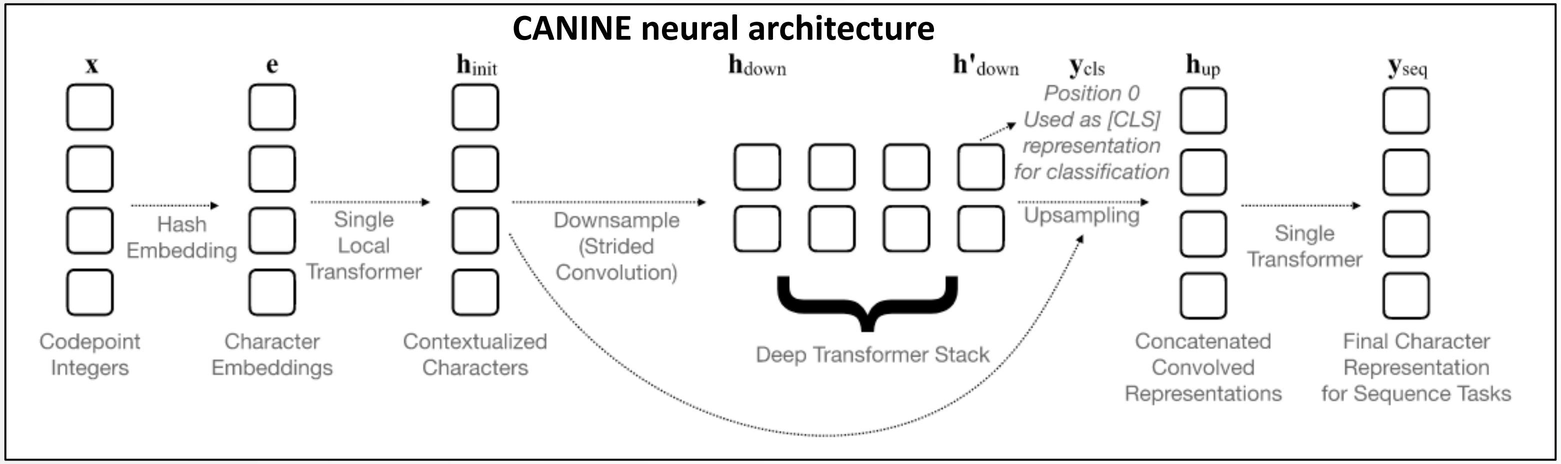
CANINE: Character Architecture with No tokenization In Neural Encoders.

- CANINE is a large language encoder with a deep transformer stack at its core.
- Inputs to the model are sequences of Unicode characters. 143,698 Unicode codepoints assigned to characters covers 154 scripts and over 900 languages!
- To avoid slowdown from increasing sequence length, it uses stride convolutions to down-sample input sequences to a shorter length, before the deep transformer stack to encode context.

- Three primary components:
  - Vocab free embedding technique;
  - Character-level model (CLM) with efficiency measures (up/down sampling of sequences); and
  - Perform unsupervised masked LM (MLM) pretraining on the CLM using variants:
    - Autoregressive character prediction
    - Subword prediction

Clark et al. "CANINE: Pre-training an efficient tokenization-free encoder for language representation." (2022).

Aside: Token free models - CANINE



- The overall functional composition form uses [UP|DOWN]-sampling, and primary encoder:

$Y_{seq} \leftarrow$ UP(ENCODE(DOWN (e))) where $e \in \mathbb{R}^{n \times d}$ is an input characters sequence, and
$Y_{seq} \in \mathbb{R}^{n \times d}$ is output of sequence predictions

- **Down-sampling**: $h_{init} \leftarrow \text{LOCALTRANSFORMER}(e)$; $h_{down} \leftarrow \text{STRIDEDCONV}(h_{init}, r)$

where $h_{down} \in \mathbb{R}^{m \times d}$ and $m = \frac{n}{r}$ is the number of downsampled positions

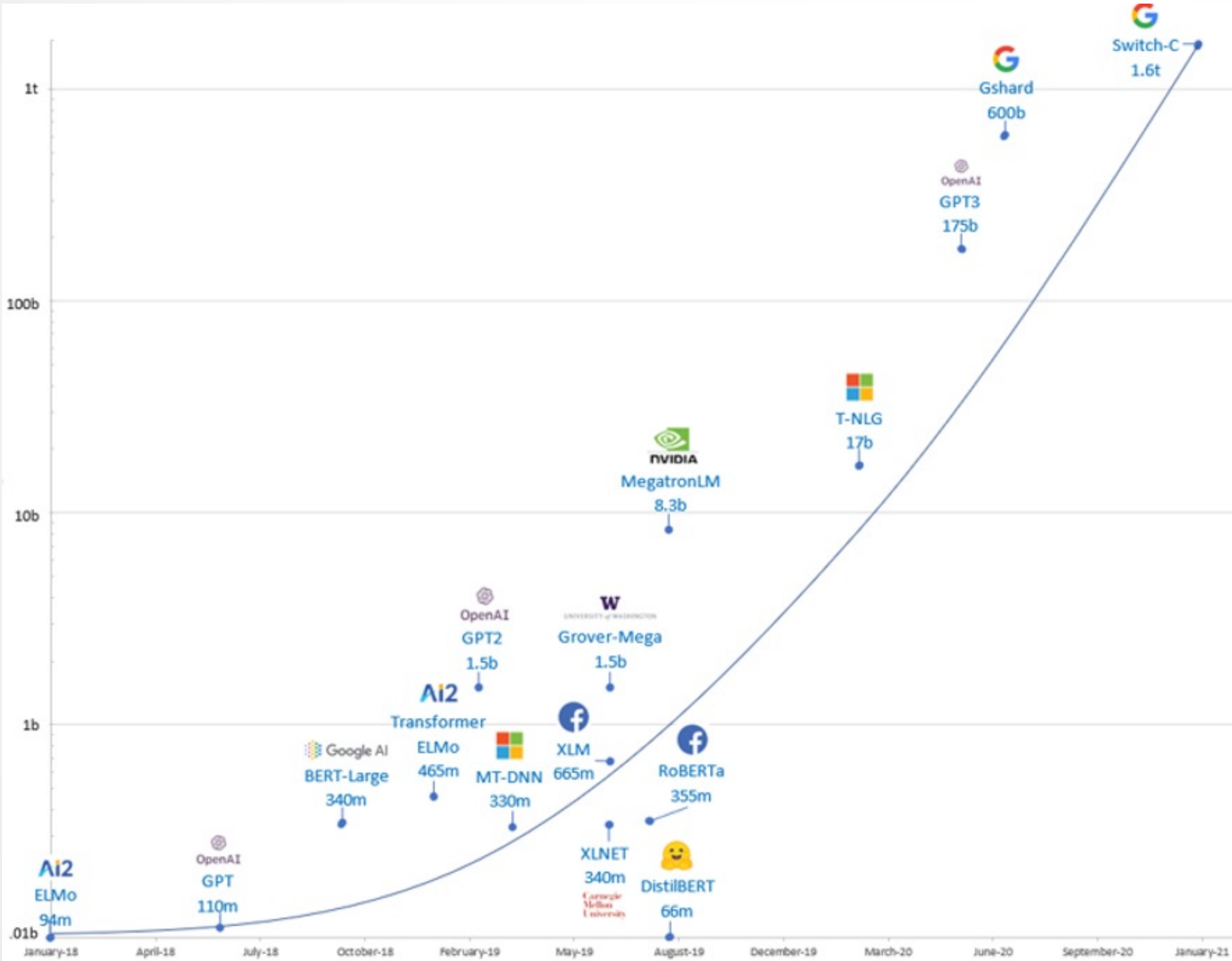
- **Up-sampling**: prediction require model's output layer sequence length to match input's length

$h_{up} \leftarrow \text{CONV}(h_{init} \oplus h'_{down}, w)$; $y_{seq} \leftarrow \text{TRANSFORMER}\left(h_{up}\right)$

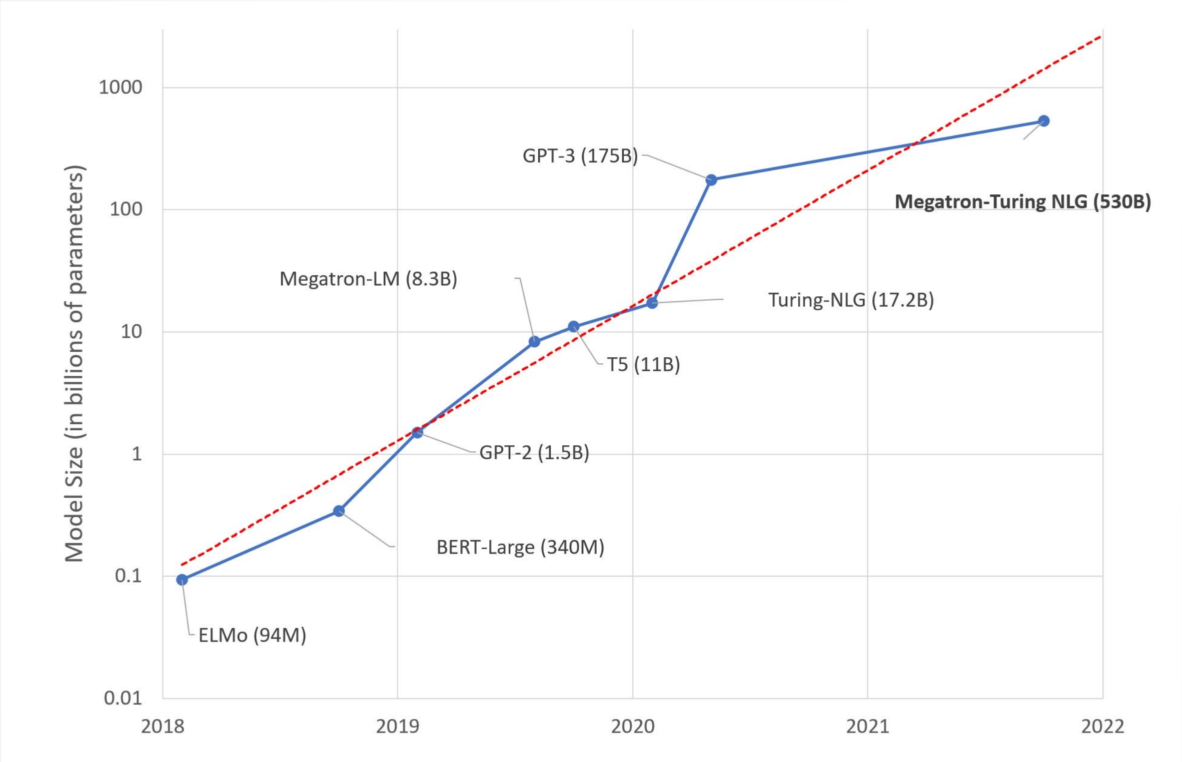
where $\oplus$ is vector concatenation, CONV projects $\mathbb{R}^{n \times 2d}$ back to $\mathbb{R}^{n \times d}$ across a window of w characters. Applying a final transformer layer yields a final sequence representation: $Y_{seq} \in \mathbb{R}^{n \times d}$

NLM TRENDS & IMPLICATIONS

NLM: the bigger is better trend



NLM: the bigger is better trend



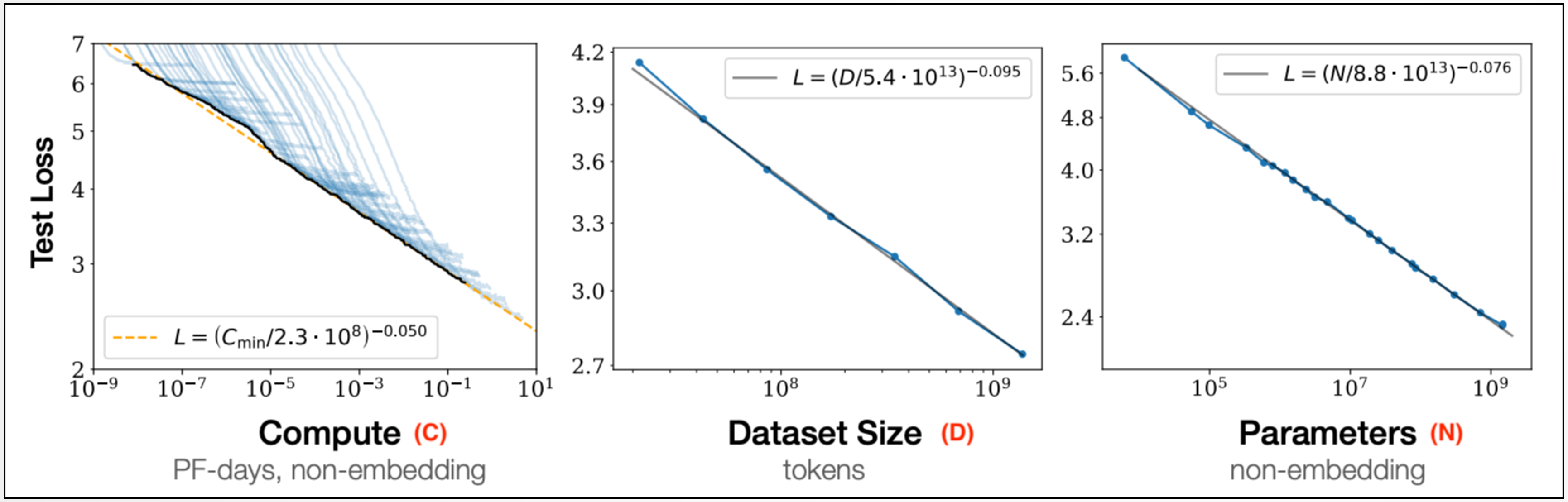
- Cons:
  - Deep learning == Deep pockets? Democratisation of compute power
  - Social impact e.g. (environmental): *"training BERT on GPU is roughly equivalent to a trans-American flight"*[1]

[1] S. Emma, A. Ganesh, and A. McCallum. "Energy and policy considerations for deep learning in NLP. (2019)" *[arxiv]*

UNIVERSITY OF TORONTO

Scaling laws for NLMs

- Kaplan et al. (2020) does a systematic review of scaling laws for NLMs [1]



Language modelling performance (decreasing test loss is better), as the factors are scaled up

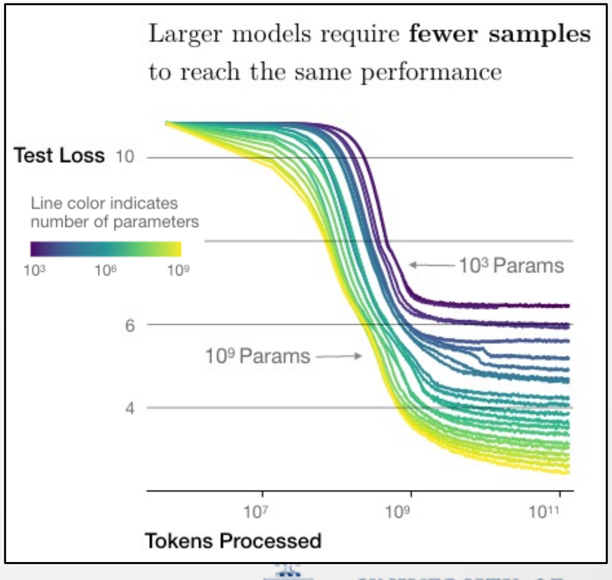
- Three **scale factors**:
  - **C**ompute: the amount of compute **C** used for training
  - **D**ataset size: the size of the dataset **D**
  - Model parameters: the number of model parameters **N**, excluding embeddings)

[1] Kaplan et al. "Scaling laws for neural language models." (2020). **link**

Scaling laws for NLMs

Key Findings: **Performance of (Transformer based) NLMs:**

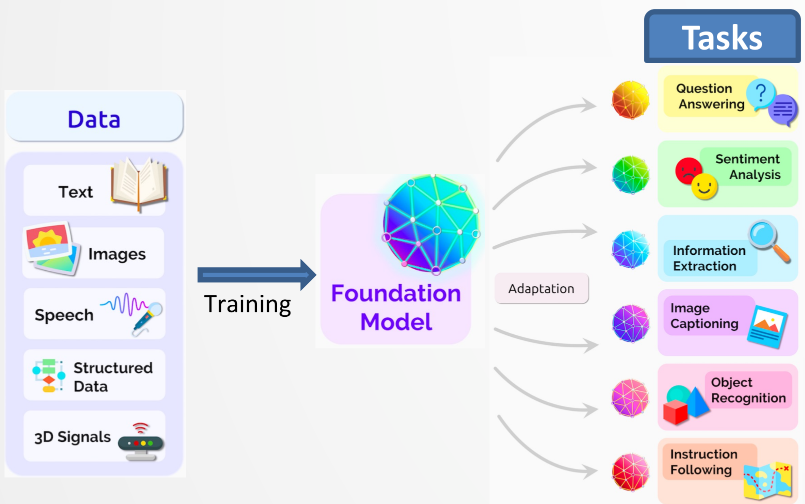
- Has power-law relationship with the three scale factors: C, D, N (excluding embeddings).
- Depends most strongly on these scale factors; architectural hyperparameters (like depth, width) does not have much effect.
- Improves smoothly when the factors (N, D) are scaled up in tandem. Diminishing returns if either N or D bottlenecks the other. Roughly, an 8x model size increase should match 5x data size increase to avoid performance penalty.
- **Transfer learning**: out-of-distribution generalization depends almost exclusively on the in-distribution (train set) validation loss performance that improves with the scaling factors.
- **Sample efficiency**: Large models are mode sample-efficient than small models, reaching the same level of performance with fewer optimization steps, data points.



[1] Kaplan et al. "Scaling laws for neural language models." (2020). **link**

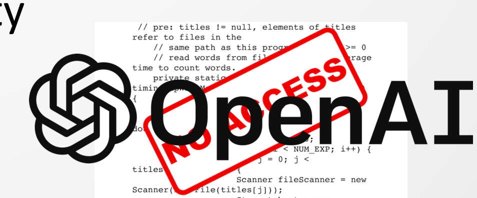
LLMs as Foundation Models

- **Homogenization**: (almost) all SOTA NLP LLM models are now adapted from one of a few foundation models (like BERT, BART, T5, etc.). [1]



- Data from various modalities
- Adoption to a wide range of downstream tasks

- **Social Impact**
  - Exacerbation of social inequalities.
  - Democratization: increased computation demands – power/capability concentrated to few corporations/start-ups.
  - Gap between industry models and community models are large.
  - Increasing proprietary moat and closed source nature.
  - Solution: *government intervention*?



[1] Bommasani, Rishi, et al. "On the opportunities and risks of foundation models." (2021). link.